



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 1998

Kompression drei- und vierdimensionaler medizinischer Bilddaten

Steingötter, A ; Werner, C D ; Sachse, F B ; Dössel, O

DOI: <https://doi.org/10.1515/bmte.1998.43.s1.478>

Other titles: Compression of 3- and 4-dimensional medical image data

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-58607>

Journal Article

Published Version

Originally published at:

Steingötter, A; Werner, C D; Sachse, F B; Dössel, O (1998). Kompression drei- und vierdimensionaler medizinischer Bilddaten. Biomedizinische Technik. Biomedical engineering, 43(s1):478-479.

DOI: <https://doi.org/10.1515/bmte.1998.43.s1.478>

Kompression drei- und vierdimensionaler medizinischer Bilddaten

A. Steingötter, C. D. Werner, F. B. Sachse, O. Dössel

Institut für Biomedizinische Technik, Universität Karlsruhe, D 76128 Karlsruhe

EINLEITUNG

In der Medizin werden mit modernen tomographischen Verfahren immer häufiger drei- und vierdimensionale Bilder erzeugt. Solche Bilder umfassen ein enorm großes Datenvolumen. Ohne eine Kompression der 3D- bzw. 4D-Bilder ist ein schneller Datentransfer und eine effiziente Archivierung kaum denkbar. Hierbei muß beachtet werden, daß zur Diagnose verwendete Bilddaten aus rechtlichen Gründen nicht mehr verändert werden dürfen. Dies bedingt den Einsatz eines möglichst verlustlosen Komprimierers.

Das hier vorgestellte Kompressionsverfahren drei- und vierdimensionaler medizinischer Bilder beruht im wesentlichen auf dem für 2D-Bilder geschaffenen JPEG-Standard [5]. Zur Bewertung dieses verlustbehafteten Kompressionsverfahrens haben wir zusätzlich die beiden verlustlosen Runlength- und Huffman-Codierungsmethoden für 3D- und 4D-Bilddaten implementiert.

KOMPRESSION

Im allgemeinen lassen sich Kompressionsverfahren in zwei Kategorien aufteilen. Zum einen in die Kategorie verlustloser (lossless) und zum anderen in die der verlustbehafteten (lossy) Verfahren [3][1]. Verlustlose Kompressionsmethoden sind vollständig reversibel, d. h. aus den komprimierten Daten lassen sich die Ursprungsdaten vollständig zurückgewinnen. Bei den verlustbehafteten Methoden hingegen kommt es zu einer Datenreduktion. Der Nachteil verlustloser Komprimierer ist, daß die Kompressionsraten deutlich unterhalb derer von verlustbehafteten Komprimierern liegen.

Zwei verlustlose Kompressionsmethoden, welche in den nächsten Kapiteln noch kurz behandelt werden, sind die Runlength- und die Huffman-Codierung. Beide sind Entropie-Codierungen und bedienen sich Codes variabler Länge [3][2].

Ein verbreitetes und standardisiertes verlustbehaftetes Kompressionsverfahren ist der JPEG- (Joint Photographic Expert Group-) Standard. Er zeichnet sich durch hohe erreichbare Kompressionsraten und den für jeden frei verfügbaren Sourcecode aus.

Alle drei Kompressionsalgorithmen wurden von uns zur Komprimierung der 3D- und 4D-Bilddaten angewendet.

RUNLENGTH-ENCODING (RLE)

Die RLE, zu deutsch Lauflängen-Codierung, ist ein spezielles Verfahren zur Entropie-Codierung, welches



Abbildung 1: Der verlustbehaftete JPEG-Algorithmus

insbesondere bei zweipegeligen Signalen, z. B. '0' und '1', effizient angewendet wird. Hierbei wird die Anzahl der Abtastwerte gezählt, die denselben Pegel aufweisen. Es findet also eine Transformation vom Zweipegelsignal in ein Lauflängensignal statt. Dieses Lauflängensignal besitzt weniger Abtastpunkte, jedoch mehr Pegelstufen [3][1].

HUFFMAN-CODIERUNG

Diese Codierung beruht auf der Erstellung von Codes, deren Länge entsprechend der diskreten Wahrscheinlichkeit des Symbols, welches sie repräsentieren, variiert. Codes für Symbole niedriger Wahrscheinlichkeit besitzen eine größere Bitlänge als Codes für Symbole mit hoher Wahrscheinlichkeit. Für den Entwurf eines Huffmancodes wird meistens ein Codebaum erstellt, in dem jedes Codesymbol durch einen eindeutigen Pfad im Baum repräsentiert ist [3][2][4].

JPEG-STANDARD

Der JPEG-Standardalgorithmus zur Kompression zweidimensionaler Bilder ist ein verlustbehaftetes Kompressionsverfahren [2][5]. Er besteht im wesentlichen aus drei aufeinanderfolgenden Operationen (siehe Abb. 1). Die Diskrete Cosinus-Transformation (DCT) erfolgt hierbei über 8x8 große Bildblöcke. Sie werden einzeln und nacheinander vom Raumbereich x, y in den Frequenzraum u, v transformiert. Man erhält somit eine das gesamte Bild charakterisierende Frequenzmatrix. Die zweite Operation des Algorithmus, die Quantisierung der Frequenzmatrix, ist der eigentlich verlusterzeugende Schritt. Hierbei werden Werte von hochfrequenten irrelevanten Frequenzanteilen des Bildes mittels großer Quantisierungskoeffizienten auf Null gedrückt. Daran anschließend beginnt die verlustlose Codierung der quantisierten Frequenzkoeffizienten. Zunächst wird dazu die Frequenzmatrix zu einer Zick-Zack-Sequenz (siehe Abb. 2), umgeordnet. Auf das eindimensionale Array wird nun eine Entropie-Codierung angewendet. Die Entropie-Codierung besteht hierbei aus ei-

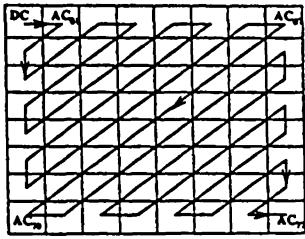


Abbildung 2: Pfad der Zick-Zack-Sequenz

ner Runlength- (Laufängen-) und einer Huffman-Codierung. Zuerst wird jeder Koeffizient der ungleich Null ist, durch zwei ihn beschreibende Symbole repräsentiert:

Symbol-1 (Laufänge, Bitanzahl) Symbol-2 (Amplitude)

Das erste Symbol besteht aus der Laufänge, welche der dem Koeffizienten vorangehenden Anzahl an Nullen entspricht, und der Bitanzahl seines Amplitudenwertes. Das zweite Symbol ist der Amplitudenwert selbst. Beide Symbole werden nun als Code variabler Länge codiert. Symbol-1 mit einem Code aus der Huffman-Tabelle und Symbol-2 mit denen für die Amplitude wirklich benötigten Bits [5].

3D-JPEG

Der JPEG-Algorithmus wurde für die Kompression dreidimensionaler Bilddaten erweitert. Die ursprüngliche Operationsabfolge blieb dabei dieselbe. Als erstes mußte an Stelle einer 2D-DCT eine 3D-DCT erfolgen. Dabei gilt, daß sich die 3D-DCT aus drei eindimensionalen DCTs in jede der drei Raumrichtungen berechnen läßt [3]. Die Berechnungsvorschrift lautet:

$$F(\vec{f}) = C_0 \sqrt{\frac{8}{d_0 d_1 d_2}} \sum_{x=0}^{d_0-1} \sum_{y=0}^{d_1-1} \sum_{z=0}^{d_2-1} P(\vec{x}) \cdot C(\vec{f}, \vec{x})$$

$$\text{mit } C(\vec{f}, \vec{x}) = \prod_{i=0}^2 \cos \left[\frac{(2x_i + 1)f_i \pi}{2d_i} \right];$$

$$C_0 = \begin{cases} \frac{\sqrt{2}}{4} & \vec{f}^T = (0, 0, 0) \\ \frac{1}{2} & \vec{f}^T = [(0, 0, a) \vee (0, a, 0) \vee (a, 0, 0)] \\ \frac{\sqrt{2}}{2} & \vec{f}^T = [(0, a, b) \vee (a, 0, b) \vee (a, b, 0)] \\ 1 & \vec{f}^T = (a, b, c) \end{cases};$$

$$a \neq 0, b \neq 0, c \neq 0; \vec{f} = \begin{pmatrix} f_0 \\ f_1 \\ f_2 \end{pmatrix}, \vec{x} = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}.$$

$P(\vec{x})$ ist der Pixelwert, d_0, d_1, d_2 geben die Ausdehnung des Datenvolumens an.

Die 3D-DCT wird auf $4 \times 4 \times 4$ große Würfel des 3D-Bildes angewendet. Damit sind, wie im 2D Fall, 64 Frequenzkoeffizienten zu berechnen. Anstelle einer Quantisierungsmatrix tritt ein Quantisierungswürfel. Es werden Kugelflächen gleicher Quantisierungsstufen definiert. Alle Würfелеlemente einer solchen

Fläche haben den gleichen Abstand zum Ursprung und besitzen denselben Quantisierungswert. Die Differenz zweier aufeinanderfolgender Quantisierungsstufen ist die Quantum-Schrittweite. Mit ihr kann die resultierende Kompressionsrate bzw. Bildqualität bestimmt werden. Je größer die Quantum-Schrittweite, desto größer die Anzahl an erzeugten Nullen im Frequenzwürfel.

Die quantisierten Frequenzwerte werden, wie im 2D-Algorithmus, in ein eindimensionales Array umgeordnet. Ähnlich der Zick-Zack-Sequenz, werden die Werte entsprechend ihrem Abstand zum Ursprung geordnet. Man beginnt mit dem DC-Wert (0,0,0) und schließt im folgenden die AC-Werte der ersten Quantisierungsstufe, dann die der Zweiten, der Dritten usw. an.

Über dieses eindimensionale Array aus quantisierten Frequenzwerten findet jetzt die Entropie-Codierung statt. Das Vorgehen unterscheidet sich im weiteren nicht mehr von dem im zweidimensionalen Fall.

4D-JPEG

Zur Kompression der 4D-Bilddaten wird eine weitere Dimension zur DCT-Berechnung hinzugefügt. Transformiert wird über eine $4 \times 4 \times 4 \times 4$ Matrix. Man erhält hierbei 256 Frequenzkoeffizienten. Die benötigte Bitanzahl wird für den neuen Frequenzwertebereich erweitert. Die Koeffizienten werden wieder nach ihrem Abstand vom Ursprung in ein 1D-Array umgeordnet.

ERGEBNISSE

Die Kompressionsraten, die durch die hier vorgestellten Kompressionsverfahren erreicht werden, hängen extrem von den Datensätzen ab. Für 3D- und 4D-MR-Datensätze des Thorax ergeben sich Raten bei RLE- und Huffman-Kompression von ca. 1 : 2,5, bei 3D-JPEG-Kompression mit maximaler Bildqualität von ca. 1 : 3 bei höchstmöglicher Qualität. Gewebeklassifizierte Daten lassen sich mit deutlich höherer Rate (ca. 1 : 30) komprimieren.

LITERATUR

- [1] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley, Reading, Massachusetts; Menlo Park, California; New York; Bonn, 1992.
- [2] Mark Nelson. *Datenkomprimierung*. Heinz Heise GmbH & Co KG, Hannover, 1. Auflage, 1993.
- [3] Jens-Rainer Ohm. *Digitale Bildcodierung*. Springer, Berlin; Heidelberg; New York, 1995.
- [4] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2. Auflage, 1994.
- [5] Gregory K. Wallace. The JPEG Still Picture Compression Standard. *Communication of the ACM*, 34(4):30-44, April 1991.